AMENDMENTS TO THE CLAIMS:

This listing of claims will replace all prior versions and listings of claims in the application:

1. (Currently amended) A method executed in a data processing system for providing communication access between a first process and a second process, the method comprising ~~the steps, executed in a data processing system, of~~:

appending security context information for the first process in a process table;

opening a socket between the first process and the second process; ~~and~~

transmitting a packet from the first process to the second process through the open socket, the packet comprising ~~including~~ the security context information for the first process in the process table; and

determining if the first and second process belong to a channel; and

accepting the transmitted packet when the first and second process belong to the channel.

2. (Original) The method of claim 1, further comprising modifying a socket structure so as to accept the security context information.

3. (Original) The method of claim 1, further comprising:

receiving the packet at the second process through the socket;

verifying the security context information received in the packet; and

permitting use of the packet if the security context information is verified.

4. (Canceled)

5. (Currently amended) The method of claim [[4]] 3, wherein determining if the first and second process belong to a channel comprises ~~includes~~:

comparing the security context information in the received packet and security context information in another process table.

6. (Original) The method of claim 5, wherein the process table and the another process table are located on a single node.

7. (Currently amended) The method of claim 3, wherein verifying the security context information comprises ~~includes~~:

determining whether the first and second process belong to two different linked channels; and

permitting use of the packet when the different channels are linked.

8. (Currently amended) The method of claim 7, wherein determining whether the first and second process belong to two different linked channels comprises ~~includes~~:

initiating a process that spawns two child processes that are connected by a shared-memory region in a memory.

9. (Currently amended) The method of claim 7, wherein permitting use of the packet comprises ~~includes~~:

decrypting the packet on a node; and

authenticating a sender associated with the first process on the node.

10. (Currently amended) The method of claim 1, wherein appending security context information comprises ~~includes~~:

obtaining the security context information from a third process, the security context information comprising ~~including~~ a virtual address and a node identification; and limiting each of the first, second and third processes to communicate with another process provided that the communicating processes share the same node identification.

11. (Original) The method of claim 1, further comprising:

modifying a network stack such that the network stack requires the security context information to be present in the socket structure to transmit.

12. (Currently amended) A method for placing processes executed in a node in a security context, comprising ~~the steps of~~:

sending a request from the node to a server to verify a username and a node identification associated with a process;

in response to the request, receiving security context information at the node from the server, the security context information comprising ~~including~~ a virtual address for the node;

initiating the process; and

appending the security context information and the node identification associated with the process in a process table.

13. (Currently amended) The method of claim 12, wherein receiving security context information further comprises ~~includes~~:

receiving a key that corresponds to the node identification from the server.

14. (Currently amended) The method of claim 13, further comprising:

encrypting a packet transmitted by the process using the key;

encapsulating the encrypted packet with a header that comprises ~~includes~~ the node identification.

15. (Currently amended) The method of claim 12, further comprising:

sending a second request from the node to the server to verify a username and node identification;

receiving additional security context information from the server, wherein the additional security context information comprises ~~includes~~ a second virtual address for the node;

creating a second process; and

appending the security context information for the second process in the process table that is associated with the second process.

16. (Currently amended) A method executed in a data processing system for providing secure communications between a first process and a second process, the method comprising ~~the steps, executed in a data processing system, of~~:

obtaining a node identification and a virtual address;

including the node identification and the virtual address in a field corresponding to the first process in a process table;

transmitting a datagram that contains the node identification and the virtual address from the first process to a socket; and

receiving the datagram at the second process that contains the node identification and a second virtual address.

17. (Currently amended) The method of claim 16, wherein obtaining a node identification and a virtual address further comprises ~~includes~~:

modifying a socket structure in the socket so that the socket structure accepts the node identification and the virtual address; and modifying a process table so that the table comprises ~~includes~~ a node identification field and a virtual address field.

18. (Currently amended) A system for providing communication access between a first process and a second process, comprising:

means for appending security context information for the first process in a process table;

means for opening a socket between the first process and the second process; ~~and~~

means for transmitting a packet from the first process to the second process through the open socket, the packet comprising ~~including~~ the security context information for the first process in the process table;

means for determining if the first and second process belong to a channel; and

means for accepting the transmitted packet when the first and second process belong to the channel.

19. (Original) The system of claim 18, further comprising means for modifying a socket structure so as to accept the security context information.

20. (Original) The system of claim 18, further comprising:

means for receiving the packet at the second process through the socket;

means for verifying the security context information received in the packet; and

means for permitting use of the packet if the security context information is verified.

21. (Canceled)

22. (Currently amended) The system of claim [[21]] 20, wherein means for determining if the first and second process belong to a channel comprises ~~includes~~:

means for comparing the security context information in the received packet and security context information in another process table.

23. (Original) The system of claim 22, wherein the process table and the another process table are located on a single node.

24. (Currently amended) The system of claim 20, wherein means for verifying the security context information comprises ~~includes~~:

means for determining whether the first and second process belong to two different linked channels; and

means for permitting use of the packet when the different channels are linked.

25. (Currently amended) The system of claim 24, wherein means for determining whether the first and second process belong to two different linked channels comprises ~~includes~~:

means for initiating a process that spawns two child processes that are connected by a shared-memory region in a memory.

26. (Currently amended) The system of claim 24, wherein means for permitting use of the packet comprises ~~includes~~:

means for decrypting the packet on a node; and

means for authenticating a sender associated with the first process on the node.

27. (Currently amended) The system of claim 18, wherein means for appending security context information comprises ~~includes~~:

means for obtaining the security context information from a third process, the security context information comprising ~~including~~ a virtual address and a node identification; and

means for limiting each of the first, second and third processes to communicate with another process provided that the communicating processes share the same node identification.

28. (Original) The system of claim 18, further comprising:

means for modifying a network stack such that the network stack requires the security context information to be present in the socket structure to transmit.

29. (Currently amended) A system for placing a process executed in a node in a security context, comprising:

a server; and

a sending node comprising:

a transmission module that transmits a request to the server to verify a user name and a node identification, and receives security context information from the server in response to the request, wherein the security context information comprises ~~includes~~ a virtual address for the sender node;

memory containing a process and an associated process table; and

an appending module that appends the received security context information and the node identification for the process in the process table.

30. (Original) The system of claim 29, wherein the transmission module further receives a key that corresponds to the node identification from the server.

31. (Currently amended) The system of claim 30, further comprising:

an encryption module that encrypts a packet transmitted by the process using the key;

an encapsulating module that encapsulates the encrypted packet with a header that comprises ~~includes~~ the node identification.

32. (Currently amended) The system of claim 29, further comprising:

a gateway that provides communication between the process and a second process executing in the node, and

wherein the transmission module further sends a second request to the server to verify a username and node identification, and receives additional security context information from the server, wherein the additional security context information comprises ~~includes~~ a second virtual address for the node;

appending the security context information for the second process in a process table that is associated with the second process.

33. (Original) A system for providing secure communications between a first process and a second process, comprising:

means for obtaining a node identification and a virtual address;

means for including the node identification and the virtual address in a field corresponding to the first process in a process table;

means for transmitting a datagram that contains the node identification and virtual address from the first process to a socket; and

means for receiving the datagram at the second process that contains the node identification and a second virtual address.

34. (Currently amended) The system of claim 33, wherein means for obtaining a node identification and a virtual address further comprises:

means for modifying a socket structure in the socket so that the socket structure accepts the node identification and the virtual address; and

means for modifying a process table so that the table comprises ~~includes~~ a node identification field and a virtual address field.

35. (Currently amended) A computer readable medium for controlling a data processing system to perform a method for providing communication access between a first process and a second process, comprising:

an appending module for appending security context information for the first process in a process table;

an opening module for opening a socket between the first process and the second process; ~~and~~

a transmitting module for transmitting a packet from the first process to the second process through the open socket, the packet comprising ~~including~~ the security context information for the first process in the process table;

a determining module for determining if the first and second process belong to a channel; and

an accepting module for accepting the transmitted packet when the first and second process belong to the channel.

36. (Original) The computer readable medium of claim 35, further comprising a modifying module for modifying a socket structure so as to accept the security context information.

37. (Original) The computer readable medium of claim 35, further comprising:

a receiving module for receiving the packet at the second process through the socket;

a verifying module for verifying the security context information received in the packet; and

a permitting module for permitting use of the packet if the security context information is verified.

38. (Canceled)

39. (Currently amended) The computer readable medium of claim 38, wherein the determining module comprises ~~includes~~:

a comparing module that compares the security context information in the received packet and security context information in another process table.

40. (Original) The computer readable medium of claim 39, wherein the process table and the another process table are located on a single node.

41. (Currently amended) The computer readable medium of claim 37, wherein the verifying module comprises ~~includes~~:

a determining module for determining whether the first and second process belong to two different linked channels; and

a permitting module for permitting use of the packet when the different channels are linked.

42. (Currently amended) The computer readable medium of claim 41, wherein the determining module comprises ~~includes~~ a initiating module that initiates a

process that spawns two child processes that are connected by a shared-memory region in a memory.

43. (Currently amended) The computer readable medium of claim 41, wherein the permitting module comprises ~~includes~~:

a decrypting module for decrypting the packet on a node; and

an authenticating module for authenticating a sender associated with the first process on the node.

44. (Currently amended) The computer readable medium of claim 35, wherein the appending module comprises ~~includes~~:

an obtaining module for obtaining the security context information from a third process, the security context information comprising ~~including~~ a virtual address and a node identification; and

a limiting module for limiting each of the first, second and third processes to communicate with another process provided that the communicating processes share the same node identification.

45. (Original) The computer readable medium of claim 35, further comprising:

a modifying module for modifying a network stack such that the network stack requires the security context information to be present in the socket structure to transmit.